데이터

LPGA2008 상금순위 40등 UP, 나머지는 DN 이분류 -> 목표변수

설명변수 : 비거리, 페어웨이, 그린적중율, 퍼팅개수, 샌드_수, 샌드 세이브

```
import pandas as pd
df=pd.read_csv('http://wolfpack.hnu.ac.kr/Stat_Notes/example_data/lpga2008.csv')
df.columns=['golfer','drive','fairway','green','putting','sand_no','sand_save','money','play_no']
df['rank']=df['money'].rank(method='min',ascending=False)
df['group']=['up' if x<=40 else 'dn' for x in df['rank']]
df.set_index('golfer',inplace=True)
df.head(3)
```

golfer	drive	fairway	green	putting	sand_no	sand_save	money	play_no	rank	group
Ahn, Shi Hyun	249.4	64.6	61.2	27.44	1.10	34.5	6063	50	49.0	dn
Alfredsson, Helen	253.8	62.7	68.2	29.36	0.66	38.8	19343	74	5.0	up
Ammaccapane										

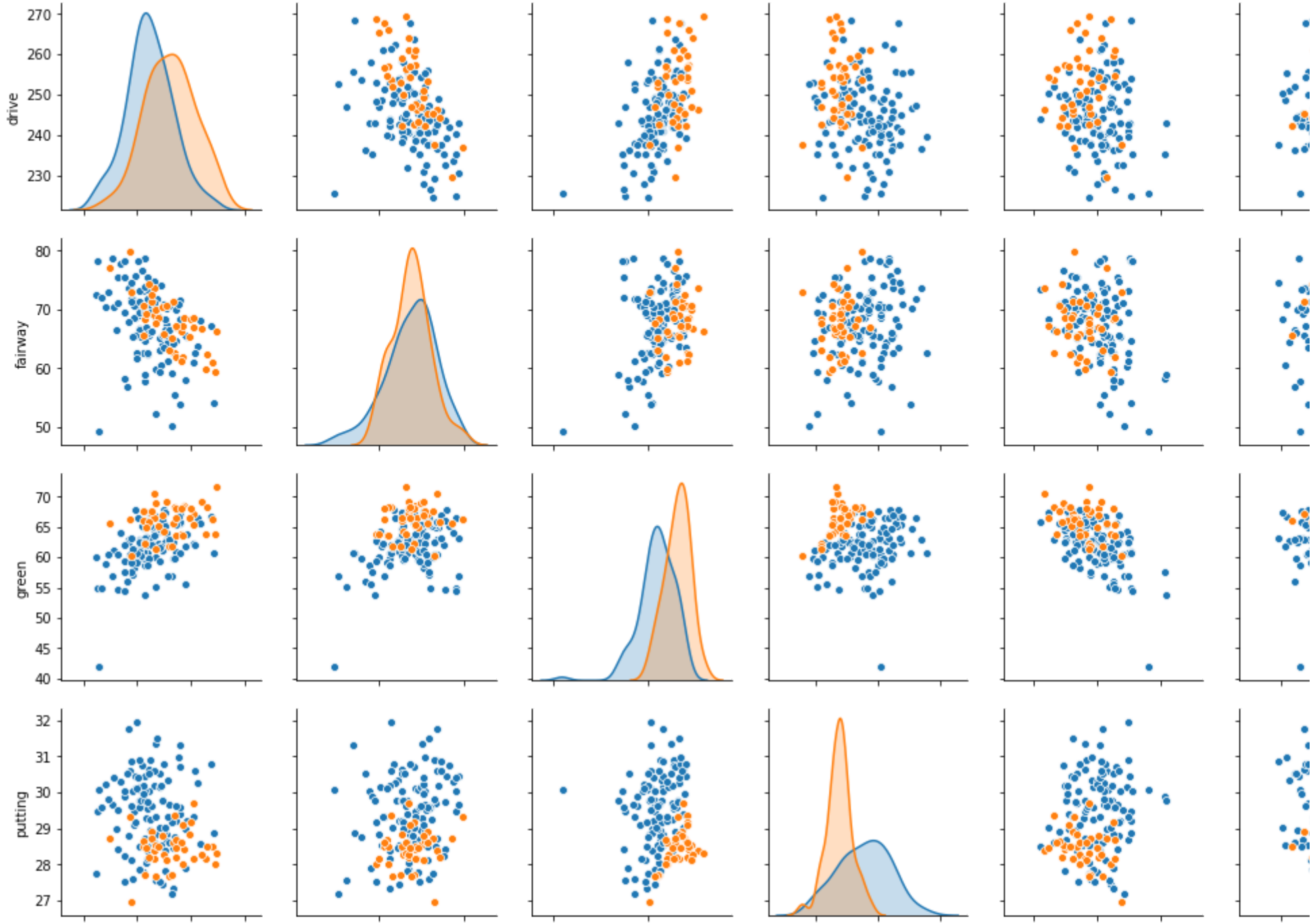
```
df.drop(['money','play_no','rank'],axis=1,inplace=True)
df.head(3)
```

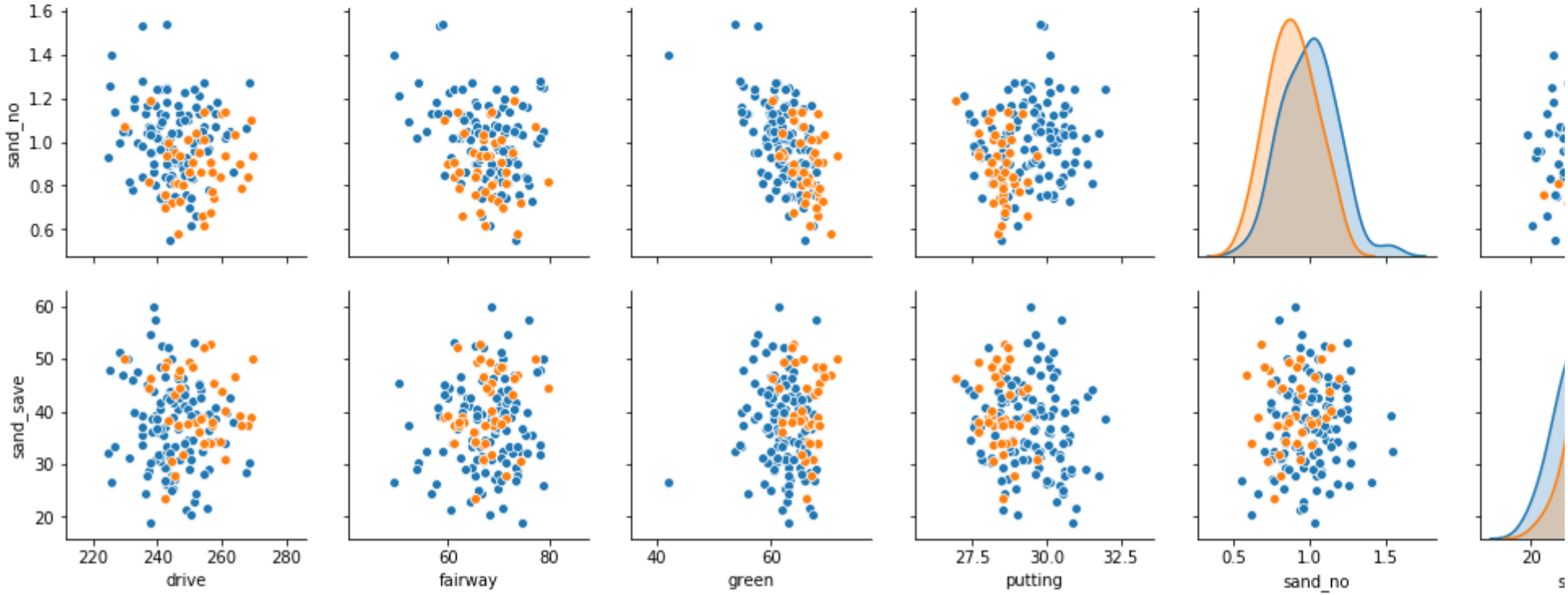
golfer	drive	fairway	green	putting	sand_no	sand_save	group
Ahn, Shi Hyun	249.4	64.6	61.2	27.44	1.10	34.5	dn
Alfredsson, Helen	253.8	62.7	68.2	29.36	0.66	38.8	up
Ammaccapane, Dina	246.3	70.2	64.6	30.20	0.74	40.5	dn

그래프 표현

```
1  import seaborn as sns
2  sns.pairplot(df, hue='group')
```

```
<seaborn.axisgrid.PairGrid at 0x7f5af3a9b048>
```





숫자 요약

```
1  df.groupby('group').mean()
```

group	drive	fairway	green	putting	sand_no	sand_save
dn	244.752137	67.547009	61.84188	29.452479	0.999573	37.051282
up	252.440000	67.667500	66.10250	28.440000	0.887250	40.592500

로지스틱 회귀분석

```
from sklearn.linear_model import LogisticRegression
X=df.iloc[:,0:6].values
y=df.iloc[:,6].values
logit_fit=LogisticRegression(fit_intercept=True).fit(X, y)
```

```
/usr/local/lib/python3.6/dist-packages/sklearn/linear_model/logistic.py:432: FutureWarning: Default solver will be c
  FutureWarning)
```

모형 추정결과

```
import statsmodels.formula.api as smf
formula='group~'
model=smf.glm(formula='group~drive+fairway+green+putting+sand_no+sand_save',
    data=df, family=sm.families.Binomial())
result=model.fit()
print(result.summary())
```

```
                 Generalized Linear Model Regression Results
==============================================================================
Dep. Variable:     ['group[dn]', 'group[up]']   No. Observations:                  157
Model:                                    GLM   Df Residuals:                      150
Model Family:                        Binomial   Df Model:                            6
Link Function:                          logit   Scale:                          1.0000
Method:                                  IRLS   Log-Likelihood:                    nan
Date:                       Sun, 13 Oct 2019   Deviance:                          nan
Time:                               09:06:27   Pearson chi2:                     45.1
No. Iterations:                          100
Covariance Type:                    nonrobust
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept    -24.3045     29.589     -0.821      0.411     -82.298      33.689
drive          0.0062      0.087      0.071      0.943      -0.164       0.177
fairway        0.1027      0.154      0.667      0.505      -0.199       0.405
green         -2.1473      0.538     -3.993      0.000      -3.201      -1.093
putting        5.7770      1.432      4.034      0.000       2.970       8.584
sand_no       -4.7350      3.560     -1.330      0.183     -11.712       2.242
sand_save     -0.1653      0.063     -2.634      0.008      -0.288      -0.042
==============================================================================
/usr/local/lib/python3.6/dist-packages/statsmodels/genmod/families/family.py:890: RuntimeWarning: invalid value enco
  n_endog_mu = self._clean((1. - endog) / (1. - mu))
/usr/local/lib/python3.6/dist-packages/statsmodels/genmod/families/family.py:942: RuntimeWarning: divide by zero enc
  special.gammaln(n - y + 1) + y * np.log(mu / (1 - mu)) +
/usr/local/lib/python3.6/dist-packages/statsmodels/genmod/families/family.py:943: RuntimeWarning: divide by zero enc
  n * np.log(1 - mu)) * var_weights
/usr/local/lib/python3.6/dist-packages/statsmodels/genmod/families/family.py:943: RuntimeWarning: invalid value enco
  n * np.log(1 - mu)) * var_weights
```

정분류 판별비율

```
y_pred=logit_fit.predict(X)
print('Accuracy : {:.2f}'.format(logit_fit.score(X, y)))
```

```
Accuracy : 0.93
```

> y_pred 에는 각 선수의 판별 집단에 들어감

```
y_pred[0:3] #첫 3명 예측 판별 집단
```

```
array(['dn', 'up', 'dn'], dtype=object)
```

집단 소속 예측확률

```
logit_fit.predict_proba(X)[0:3]
```

```
array([[0.69122326, 0.30877674],
       [0.12524459, 0.87475541],
       [0.99001999, 0.00998001]])
```

> 첫번째 선수 dn소속 확률 69%, up소속 확률 31% - 결론적으로 dn에 속함 (위의 결과 참고)

판별 결과 상세 출력

```
from sklearn.metrics import classification_report
print(classification_report(y, y_pred))
```

```
              precision    recall  f1-score   support

          dn       0.94      0.97      0.95       117
```

데이블 작성

```
1  table=pd.crosstab(df.group,y_pred)
2  table
```

col_0	dn	up
group		
dn	113	4
up	7	33

```
1  table.apply(lambda r: r/r.sum(), axis=1)
```

col_0	dn	up
group		
dn	0.965812	0.034188
up	0.175000	0.825000

새로운 선수 집단 확인

비거리=260, ..., 샌드세이브 비율 40%인 신인 선수는 상금 랭킹 집단 예측결과

```
1  new=pd.DataFrame([260,70,65,28,1.5,40]).T
2  print(logit_fit.predict_proba(new),'\n 판별집단',logit_fit.predict(new))
```

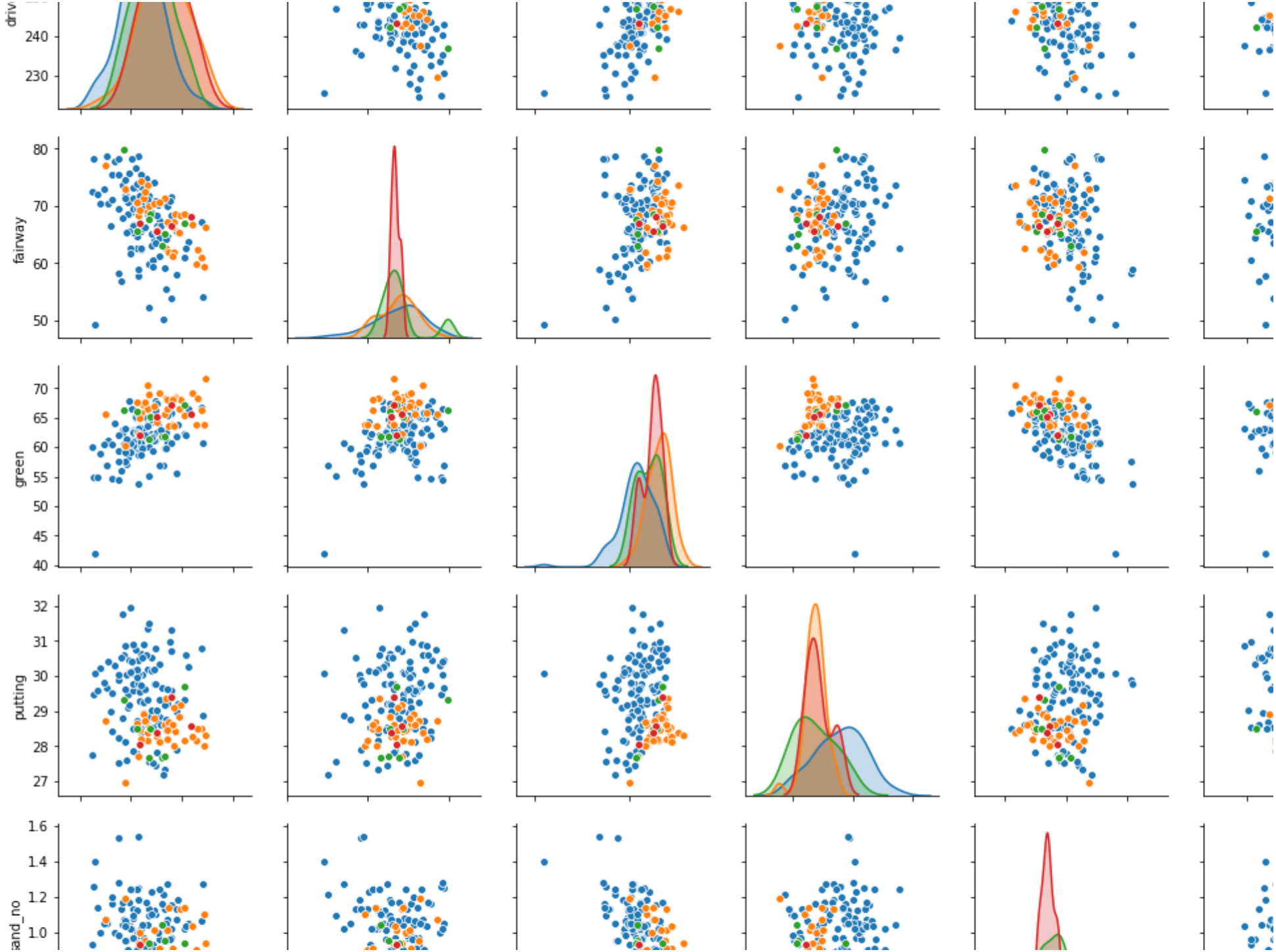
```
[[0.07041472 0.92958528]]
 판별집단 ['up']
```

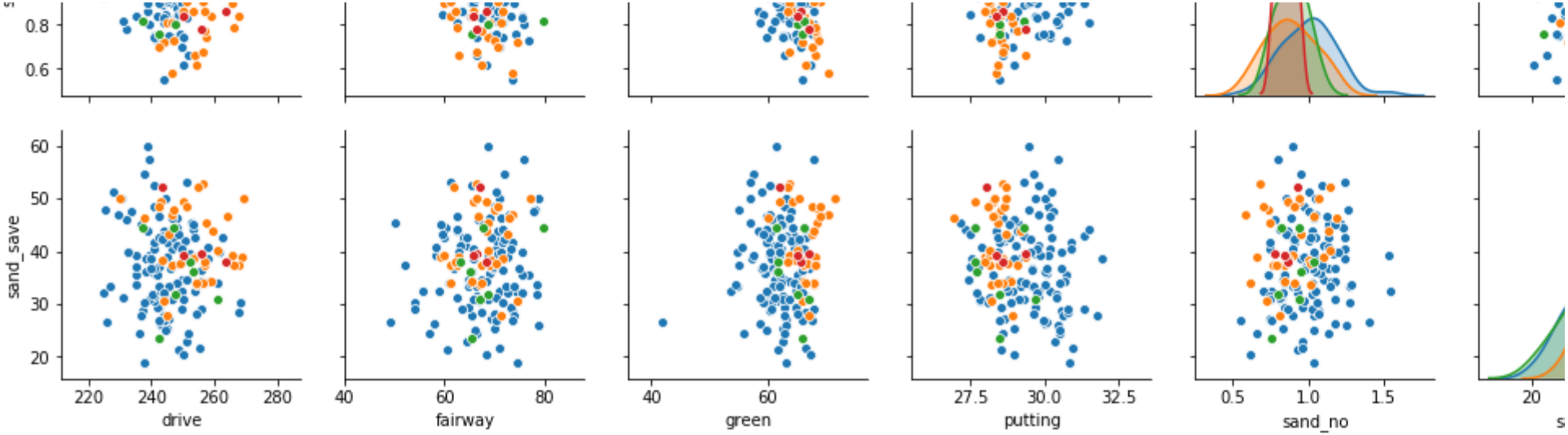
판별 결과 보기

```
y_pred=pd.DataFrame(logit_fit.predict(X))
y_pred.columns=['D_group']
y_pred.set_index(df.index,inplace=True)
df_fin=pd.concat([df,y_pred],axis=1)
df_fin['result']=df_fin.group+'-'+df_fin.D_group
df_fin.head(3)
```

golfer	drive	fairway	green	putting	sand_no	sand_save	group	D_group	result
Ahn, Shi Hyun	249.4	64.6	61.2	27.44	1.10	34.5	dn	dn	dn-dn
Alfredsson, Helen	253.8	62.7	68.2	29.36	0.66	38.8	up	up	up-up
Ammaccapane									

```
import seaborn as sns
sns.pairplot(df_fin, hue='result')
```


driv
240
230
fairway
80
70
60
50
green
70
65
60
55
50
45
40
putting
32
31
30
29
28
27
and_no
1.6
1.4
1.2
1.0



IRIS 데이터 로지스틱 판별

```
1  import pandas as pd
2  df=pd.read_csv('http://wolfpack.hnu.ac.kr/Stat_Notes/example_data/iris.csv')
3  df.head(3)
```

	Sepal_Length	Sepal_Width	Petal_Length	Petal_Width	group
0	50	33	14	2	Setosa
1	64	28	56	22	Virginica
2	65	28	46	15	Versicolor

```
from sklearn.linear_model import LogisticRegression
X=df.iloc[:,0:4]
y=df.iloc[:,4]
logreg=LogisticRegression(solver='newton-cg',multi_class='multinomial')
logit_fit=logreg.fit(X, y)
#logit_fit=LogisticRegression(multi_class='multinomial').fit(X, y)
```

정분류 비율

```
y_pred=logit_fit.predict(X)
print('Accuracy : {:.2f}'.format(logit_fit.score(X, y)))
```

```
Accuracy : 0.98
```

> 피셔 이차판별 분석 결과 98%

모형 추정

```
logit_fit.coef_
```

```
array([[-0.03681289,  0.34225369, -0.63970952, -0.35107707],
       [ 0.13448093,  0.04398746, -0.05152723, -0.44113897],
       [-0.0976752 , -0.38624283,  0.69123098,  0.79221397]])
```

```
import statsmodels.api as st
y=df.group
X=st.add_constant(df.ix[:,0:4],prepend = False)
mdl=st.MNLogit(y,X).fit()
print(mdl.summary())
```

```
Warning: Maximum number of iterations has been exceeded.
         Current function value: 0.039662
         Iterations: 35
                           MNLogit Regression Results
==============================================================================
Dep. Variable:                   group   No. Observations:                  150
Model:                         MNLogit   Df Residuals:                      140
Method:                            MLE   Df Model:                            8
Date:                 Sun, 13 Oct 2019   Pseudo R-squ.:                  0.9639
Time:                         09:55:22   Log-Likelihood:                -5.9493
converged:                       False   LL-Null:                       -164.79
Covariance Type:             nonrobust   LLR p-value:                 7.056e-64
==============================================================================
group=Versicolor       coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
Sepal_Length        -0.8721     75.235     -0.012      0.991    -148.330     146.586
Sepal_Width         -0.2164     40.865     -0.005      0.996     -80.311      79.878
Petal_Length         1.1966     47.252      0.025      0.980     -91.416      93.809
Petal_Width          1.1357     77.449      0.015      0.988    -150.662     152.933
const               12.9929   2553.843      0.005      0.996   -4992.448    5018.433
------------------------------------------------------------------------------
group=Virginica       coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
Sepal_Length       -1.1187     75.235     -0.015      0.988    -148.577     146.340
Sepal_Width        -0.8845     40.868     -0.022      0.983     -80.984      79.215
Petal_Length        2.1395     47.255      0.045      0.964     -90.478      94.757
Petal_Width         2.9643     77.455      0.038      0.969    -148.845     154.774
const             -29.6449   2553.972     -0.012      0.991   -5035.339    4976.049
==============================================================================
/usr/local/lib/python3.6/dist-packages/ipykernel_launcher.py:3: DeprecationWarning:
.ix is deprecated. Please use
.loc for label based indexing or
.iloc for positional indexing

See the documentation here:
http://pandas.pydata.org/pandas-docs/stable/indexing.html#ix-indexer-is-deprecated
  This is separate from the ipykernel package so we can avoid doing imports until
/usr/local/lib/python3.6/dist-packages/numpy/core/fromnumeric.py:2389: FutureWarning: Method .ptp is deprecated and
  return ptp(axis=axis, out=out, **kwargs)
/usr/local/lib/python3.6/dist-packages/statsmodels/base/model.py:512: ConvergenceWarning: Maximum Likelihood optimiz
  "Check mle_retvals", ConvergenceWarning)
```

```
1  from sklearn.metrics import classification_report
2  print(classification_report(y, y_pred))
```

```
              precision    recall  f1-score   support

      Setosa       1.00      1.00      1.00        50
  Versicolor       0.98      0.96      0.97        50
   Virginica       0.96      0.98      0.97        50

    accuracy                           0.98       150
   macro avg       0.98      0.98      0.98       150
weighted avg       0.98      0.98      0.98       150
```

```
1  new=pd.DataFrame([45,30,30,15]).T
2  print(logit_fit.predict_proba(new),'\n 판별집단',logit_fit.predict(new))
```

```
[[3.53907230e-01 6.46091248e-01 1.52233241e-06]]
 판별집단 ['Versicolor']
```

> 피셔 2차 판별분석 결과 64.6%, Versicolor 였음.